CS2261 Media Device Architecture

Announcements:

* Submit files as .zip, not .rar or anything else

C programs

* No need to add scope for C in this class (public static, etc)
  + C language has scope but they work differently in C than Java
* Also don’t necessarily need a return statement like in Java
  + Double check any function that requires a return statement to ensure the compiler actually sees it
* C programs are compiled linearly, so it goes in order of what you’ve written. You can’t call something you have yet to define, like a variable or a function.
* Solution to this is **prototypes**, lines of code that act like interfaces in Java.
  + Like declaring before initializing.
  + Basic Structure:
    - ReturnType funcName(param1Type, param2Type)
* Prototypes should be at the top of the file, and then your main function
  + Or prototypes in a header file, but this isn’t important now

Mode 3

* Most simple mode
* GBA info comes from the video buffer
  + The video buffer is a specific subset of memory for the VB, located at 0x6000000
* The Display control register is located at 0x4000000
  + Each bit means something
  + Bits 0-2: display mode (mode 1, 2, 3)
    - Mode 3: 011
  + Bit 4 – page select
  + Bit 6 – sprite mode
  + Bits 8-11 (background to enable)
    - We use background 2 (bg2) located at 10
* #define REG\_DISPCTL (\*(unsigned short \*)0x4000000)
* 🡪(unsigned short \*)0x4000000) – casts the number as an address
* 🡪 (\*(unsigned short \*)0x4000000) – dereferences it and is ready to store a value at the address 0x4000000
* Later in the main function you set REG\_DISPCTL = MODE3|BG2\_ENABLE
* Mode 3 is special because every pixel gets a 15-bit RGB color
  + Total of 38,400 pixels (240x160)
* White: all the colors on (0111111111111111)
* Black: all the colors off (000000000000000)
* Each color can be quantified by red | green<<5 | blue<<10
  + If you plug in the wrong number (should be below 31 because you can’t go over 31 because each color gets only 5 bits), you start pushing out other shit
* #define COLOR(r,g,b) ((r) | ((g) << 5) | ((b) << 10))
  + The bitwise“or” | to bind the three expressions together
  + All those parentheses are there because this is a blind copy-paste.
  + Blind Copy Paste and Parenthesis:
  + When you define a macro, the compiler will directly copy and paste what you place as its function, which is why the placement of parentheses are important
  + Example
  + #define SQUARE(x) (x \* x)
    - Int a = 4
    - Int b = 3
    - Int c = SQUARE(a + b)
      * Your result will be (a + b \* a + b)
      * (3 + 4 \* 3 + 4) = ( 3 + 12 + 4) = 19
      * Desired result is 49
  + #define SQUARE(x) ((x) \* (x))
    - This is what you want to make, so that the compiler doesn’t get confused. You want parentheses around the parameters and the whole expression
    - Int a = 4
    - Int b = 3
    - Int c = SQUARE(a + b)
      * Result will be ((a+b)\*(a+b))
      * =49
* Reviewed OFFSET
* Created a C program to draw three triangles for Lab02